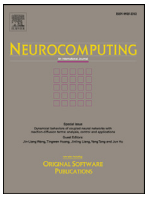
Neurocomputing 330 (2019) 259–266

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom



Semantic image segmentation via guidance of image classification

Falong Shen*, Gang Zeng

School of Electrical Engineering and Computer Science Peking University, China

ARTICLE INFO

Article history:
Received 14 May 2018
Revised 8 October 2018
Accepted 8 November 2018
Available online 16 November 2018

Communicated by Dr. Haijun Zhang

Keywords:
Semantic image segmentation
Guidance of classification
Co-occurrence

ABSTRACT

This paper describes a joint segmentation and classification approach that exploits global image features to validate the predictions from local appearance descriptors and to ensure their consistent labeling. The in-between interplay is encoded by a parameter-learning process of a unified deep learning model embedding a fully convolution network portion. Although FCN has a relatively large recept field, the integration of the image content as a whole makes the prediction more reasonable and logical, since coincidences in local neighborhoods are more likely to be depressed given global structures. We also propose a content-sensitive co-occurrence priori for label compatibility, which provides additional constraints for CRF based segmentation.

1. Introduction

Deep convolutional neural networks (CNN) has achieved impressive performance in image classification since AlexNet [1–4]. Despite the fact that CNN has been proposed since 90s [5], it becomes popular only recently thanks to the rapid development of graphic processing unit (GPU) and large scale image datasets [6]. While the secret of CNN has not been discovered clearly, it is widely adapted in many areas of computer vision and most researchers believe that the gained high performance is due to the rich information provided by deep architectures.

Image classification and semantic segmentation are two important and related topics in image understanding. Classification often relies on global features to recognize the content and treats an image as a whole, while segmentation aims at assigning pixel-wise labels and thus exploits local descriptors. Although these two tasks are defined in different scales, both require a robust and discriminative feature representation.

Deep CNN provides global high level semantic features for image classification. The task of semantic image segmentation, however, also needs to determine object position, shape and boundary, which rely on local contents. Pooling layers or strided convolution layers in CNN tend to tolerate the object translation and deformation but decrease the ability for locating and separating objects from the neighboring context.

We propose a noval semantic image segmentation model guided by image classification as shown in Fig. 2. A high abstracted semantic feature is important for semantic segmentation. It also allows to impose restrictions to the predictions raised from local recept field using FCN: The inconsistent predictions are restrained by the global understanding of whole image content. For instance, objects in different categories may share similar parts, *e.g.*, cat and dog have similar tails. It is more reasonable to make a decision by viewing them at the global level. For another example as it is shown in Fig. 1, the mid-level descriptor by FCN mistakes part of boat as that of sofa. Seeing the whole image, it is much easy to say that it is actually a boat rather than sofa.

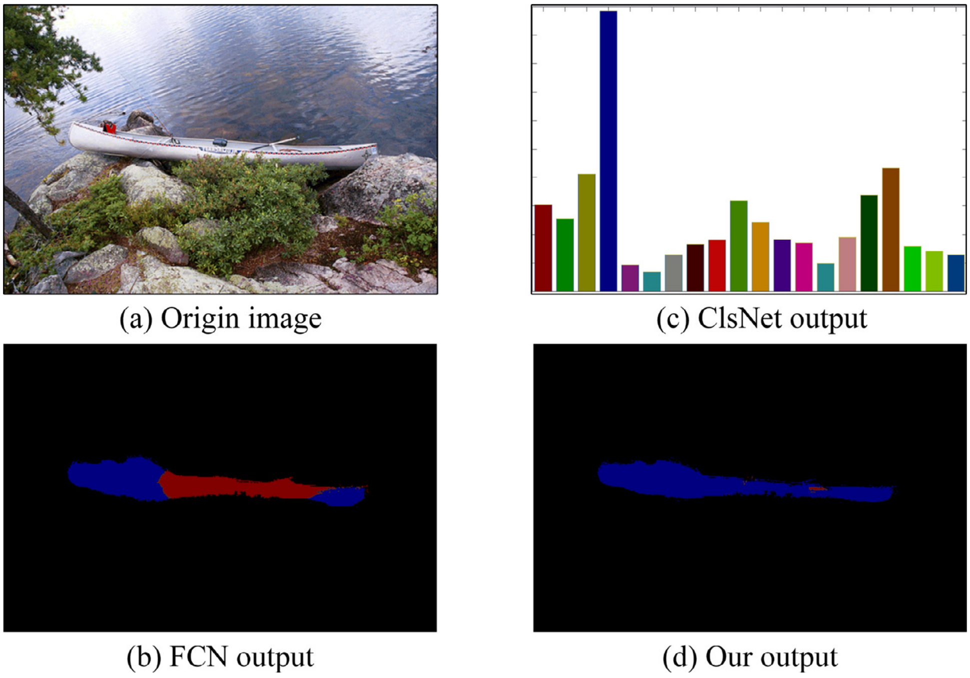
On the second aspect, the co-occurrence compatibility among different categories is also learned, which further helps to validate the compatibility of local predictions. Low penalty should be assigned to pairs of labels that tends to co-occur, such as, [person, horse], [person, bike] and [person, sofa]. Shen et al. [7] model local label compatibilities by integrating FCN with guidance CRF and train the whole deep network in an end-to-end fashion, reaching high performance on the Pascal VOC 2012 segmentation benchmark. However, the co-occurrence relationship in their model is fixed and irrelative to the content on image. We believe that this co-occurrence priori should be sensitive to the given image and thus the label compatibility should be decided based on the image content. With the content-sensitive co-occurrence priori, we employ a guidance CRF [7] to encode relationships among pixel with different colors and positions. Precise and sharp object boundaries are obtained by an efficient mean field inference algorithm.

1.1. Contribution

1. We propose a noval semantic image segmentation guided by image classification, which jointly train the parameters of both

* Corresponding author.
E-mail address: shenfalong@pku.edu.cn (F. Shen).

https://doi.org/10.1016/j.neucom.2018.11.027
0925-2312/© 2018 Elsevier B.V. All rights reserved.


(a) Origin image

(c) ClsNet output

(b) FCN output

(d) Our output

Fig. 1. The classification net (clsNet) helps FCN produce a more global consistent estimated map. The color in the histogram and segmentation map has the same mapping to categories as it is in Pascal VOC 2012.

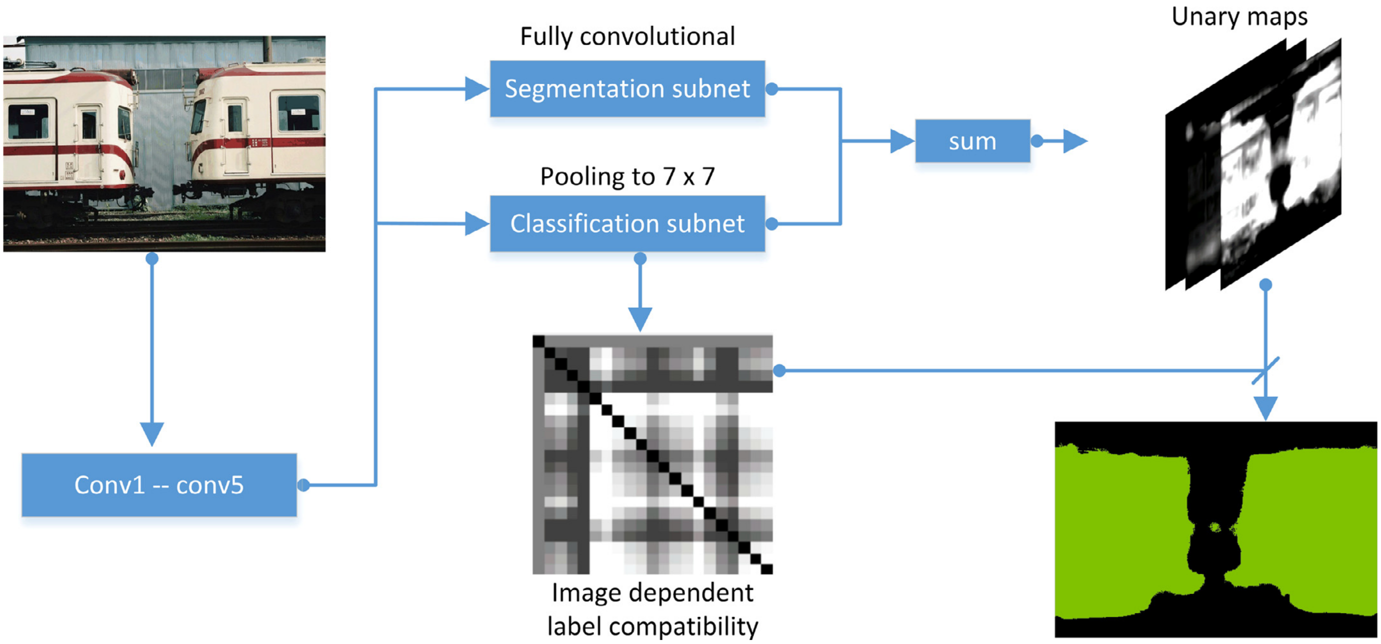


Fig. 2. An illustration of the framework of the proposed model. The given image is fed into the convolutional network to get a feature map at size 1/8 or 1/16 of original image. Two branches are followed: the first branch goes into two `Conv` layers with 1024 filters and gets a dense prediction map. In the second branch the feature map is pooled to 7 × 7 and goes into two `Inner Product` with 4096 channels to produce a global prediction. Global prediction is un-pooled to the feature map size. Finally two maps are combined yielding to the unary potential. Global prediction is also used to generate image-dependent label compatibility. Finally unary maps and label compatibility matrix are put together in a guidance CRF framework to get the final output.

the global image feature and local appearance descriptor. By integrating classification model, our method achieves a more consistent and reliable segmentation map. We evaluate our model on Pascal VOC 2012, Cityscapes and ADE-20k to demonstrate the effectiveness of our jointly learning model.

2. We exploit a content-sensitive co-occurrence priori to model label compatibility based on global image features. We use piecewise learning for relation modeling to avoid mean-field inference in each iteration.
3. In the inference stage, we run several initialization and propose a strategy to ensemble the predicted score maps, which makes our algorithm invariant to image zooming and panning operations to a certain degree.

2. Related works

Most of the state-of-the-art semantic segmentation methods [8–13] rely on a fine-tuned FCN [14,15] borrowed from a pre-trained classification model on ImageNet-1k [16,17] and a fully connected CRF [18]. Long et al. [15] firstly proposed to adopt fully convolutional networks for semantic image segmentation. On the other hand, Conditional Random Field (CRF) mainly has two objectives in semantic image segmentation, *i.e.*, (1) edge preserving, and (2) label compatibility provided by appropriate edge potentials. In the following paragraphs, we will review most related works with focus on different training method of FCN and CRF for semantic image segmentation.

Piecewise training treats CRF as a post-process disconnected with FCN in the training phase [9,11,13,19]. Starting from [19], it becomes almost a standard configuration to couple the recognition capacity of deep convolutional networks and the fine-grained localization accuracy of fully connected conditional random field in image semantic segmentation. Zheng et al. [9] trained a high performance segmentation model using part of pixel-wise annotations and part of bounding box level annotations. Noh et al. [13] proposed a deconvolutional network to identify detailed structures and handle objects in multi-scale naturally.

Generally speaking, semantic image segmentation is a standard task of structured prediction as it needs to predict labels for all the pixel in the given image. Probability graphic model, especially CRF, is suitable for structured prediction task. Original fully connected CRF models the edge potentials via low-level feature, *e.g.,* color and spatial relationship. It recovers object boundary at a detailed level and outperforms prior arts. Lin et al. [20] proposed to model the edge potential directly from FCN and achieve impressive performance. However, it costs lots of memory to save every edge potential in a fully connected CRF. The complexity is square to the number of categories. Lin et al. [20] constructed a multi-scale deep network and trained a joint potential of two nodes for each edge in the graph of the feature map directly by Deep CNN. They used a lower resolution of feature map which is 1/16 of the original image size. It once reached leading performance on most categories of the *test* set of Pascal VOC 2012.

Jointly training of FCN and CRF is another important direction in recent literatures [7,8,12,13,21–24]. Deep feature learned with structured output shows great promise in image semantic segmentation and many other domains. While it costs much time on inference in each update, jointly learning of deep features and CRF parameters leads to significant performance gains. Zheng et al. [8] and Liu et al. [10] achieved high performance using jointly learning framework. Most models of this kind are based on mean field approximation. Lin et al. [24] also proposed a method based on belief propagation [24] by directly learning the messages estimators. Fully connected CRF uses standard Potts model as label compatibility. Zheng et al. [8] proposed an end-to-end system to learn label compatibility matrix, which are image independent. Deep parsing network (DPN) proposed in [12] once reached the highest mean IoU score on the *test* set Pascal VOC 2012 by introducing additional layers to model pair-wise terms in CRF, which only relies on one mean field iteration.

3. Methods

We employ a unified model for jointly learning the global image feature for classification and the local appearance descriptor for semantic segmentation. The model is based on a deep learning framework, and it comprises a FCN component to characterize the composition of local perception and a CNN component for global scene understanding. A fine segmentation score map is then calculated using a CRF to find the Maximum A Posterior (MAP) label assignment that optimally combines the learned features, co-occurrence priori for label compatibility as shown in the Fig. 2. Finally, the additional ensemble strategy makes the proposed algorithm robust to image zooming and panning operations, which further improves its performance.

Table 1
The performance of DeepLab-MSc-LargeFOV-COCO [11] on the Pascal VOC 2012 val set.[a] With the knowledge about the object categories, this method shows a great improvement on performance and outperforms the state-of-the-art by a large margin. This simple experiment implies the importance of fusing classification knowledge for semantic segmentation.

Method	Mean IoU(%)	Per-pixel acc
w/o Gt categories	72.54	0.9328
With Gt categories	78.86	0.9486

[a] This model is available at https://bitbucket.org/deeplab/deeplab-public/. We have re-trained it on the same dataset. Note that they reported 71.7% accuracy on the val set. The gap to our result is mainly due to the fact that we run several initialization and combine these final prediction together.

3.1. Jointly learning of global and local features

The first step towards image understanding is the design of image features. We construct a novel deep structure by integrating layers for segmentation and classification, to enforce the consistency among local predictions and to enhance the recognition ability. Experiments in Table 1 validate our observation. With an extra knowledge about the object categories in an image, DeepLab model [11] can reach a very high performance on the Pascal VOC 2012 validation set, and outperforms the state-of-the-art by a large margin. We believe that the given category labels help to clear up the misunderstanding among some confusing objects (see the example in Fig. 1). While it is more easy to distinguish the confusing categories from a more global view, a too large receptive field also decreases the ability of locating object boundaries in semantic segmentation. Hence, the combination of global and local image features is a reasonable solution to the above dilemma.

A straightforward solution is a two-stage process which firstly performs classification and then uses the predicted classification labels to guide the latter semantic segmentation step. However, our preliminary experiments shows this kind of multi-label classification does not work well [25,26]. It is perhaps due to the fact that multi-label classification on Pascal VOC 2012 is still not accurate enough and once an error appears in the classification stage, it cannot be easily recovered in the latter semantic segmentation step. This motivates us to design a unified model for jointly learning of global and local features.

Let $\mathbf{x}$ and $\mathbf{y}$ denote a given image and its pixel-wise label respectively. The negative log-likelihood is characterized by a Gibbs distribution

$$-\log P(\mathbf{y}|\mathbf{x}) = \sum_{c} \phi_c(\mathbf{y}_c, \mathbf{x}_c) + Z(\mathbf{x}), \tag{1}$$

where c is a clique on graph $\mathcal{G}$ and $\phi_c(\mathbf{y}_c, \mathbf{x}_c)$ is the potential defined on the assignment. $Z(\mathbf{x}) = \log \sum_{\mathbf{y}} \exp^{\sum_c \phi_c(\mathbf{y}_c, \mathbf{x}_c)}$ is the partition function. We only take the first order cliques and second order cliques into consideration,

$$\sum_{c} \phi_c(\mathbf{y}_c|\mathbf{x}) = \sum_{i \in u} \phi(y_i|\mathbf{x}) + \lambda \sum_{(i,j) \in v} \psi(y_i, y_j|\mathbf{x}), \tag{2}$$

where $\phi(y_i|\mathbf{x})$ is the unary potential, $\psi(y_i, y_j|\mathbf{x})$ is pairwise potential and the node i and node j are in the same second order clique v. Global clues are necessary to form the unary potential. However, as stated in the previous paragraphs, it is inappropriate to treat image classification as a pre-processing step before semantic segmentation. In order to jointly train features for these two targets, we firstly divide the unary potential into two parts

$$\phi(y_i|\mathbf{x}) = \phi_l(y_i|x_i) + \phi_g(y_i \in \mathbf{L}_x|\mathbf{x}), \tag{3}$$

where $\phi_l(y_i|x_i)$ is the potential produced by FCN based on local receptive field. By enlarging the recept field of x_i and adjusting the input stride [19], we found that too large receptive field decreases the performance of pixel-wise labeling and that hierarchical information is a better choice.

Under the Deep CNN jointly learning framework, we refine local potential with the global potential $\phi_g(y_i \in \mathbf{L}_x|\mathbf{x})$ based on the whole image content $\mathbf{x}$. While $\phi_l(y_i|x_i)$ provides mid-level descriptors for segmentation, $\phi_g(y_i \in L_x|\mathbf{x})$ provides high-level information about the existing object categories in the image.

As shown in Fig. 2, the input image goes through five convolutional layers and decreases to a resolution which is 1/8 or 1/16 of original image size. We pad the image size to a multiply of 8 or 16 for computation convenience. Then the 512 channels feature map goes into segmentation subnet and classification subnet individually. For the segmentation map, we follow the large field view of Deeplab [19] which is very efficient with high performance. For the classification subnet, the whole feature map is pooled into 7 × 7 feature maps using max pooling operation, which is then fed into two `Inner Product` layers with 4096 channels. The `Inner Product` layer is much faster than convolutional layer with same channels and our net suffers little speed loss during both training and testing compared to Deeplab [19]. The output of the classification subnet is 512 channels and is duplicated to the same size of segmentation feature map. The two maps are added together and then up-sampled to the original image size.

3.2. Label compatibility

In order to address the pair-wise term in the model, we follow the typical fully connected CRF model proposed in [7], expressed in the following form

$$\psi(y_i, y_j|\mathbf{x}) = \mu(y_i, y_j|\mathbf{x})k(f_i, f_j), \tag{4}$$

where $k(f_i, f_j)$ is the kernel based on low-level features f_i and f_j (*e.g.*, RGB color, $x - y$ coordinates within the image) on pixel i and pixel j. Following the previous works [7], the kernel is in the form of

$$k(f_i, f_j) = \frac{1}{|\omega|^2}\sum_k \left[1 + (\Sigma_k + \epsilon U)^{-1}\sum_{c=1}^{3}(I_i^c - \mu_i^c)(I_j^c - \mu_j^c))\right], \tag{5}$$

where μ_k and Σ_k is the mean and 3 × 3 covariance matrix of image I in window ω_k, U is a 3 × 3 identity matrix and $|\omega|$ is the number of pixels in ω_k. ϵ is a regularized parameter and we set it to 10 throughout our experiments. We take this form in order to accelerate the message passing steps of mean field [7].

For Potts model, the label compatibility matrix $\mu(y_i, y_j|\mathbf{x}) = \delta(y_i = y_j)$ and $\delta(\cdot)$ is an indicator function, which equals 1 if the input expression is true and 0 otherwise. We propose a formation different from the existing method of Potts model [18] and end-to-end CNN-RNN system introduced by Zheng et al. [8], which jointly trains CNN with CRF and learns the kernel-dependent label compatibility matrix. In contrast, we model the label compatibility based on the image content. The label compatibility matrix is defined as

$$\mu(y_i, y_j|\mathbf{x}) = \delta(y_i = y_j) + \lambda\mu_2(y_i, y_j|\mathbf{x}), \tag{6}$$

where $\mu_2(y_i, y_j|\mathbf{x})$ is dependent on the image content

$$\mu_2(y_i, y_j|\mathbf{x}) \approx -\phi_g(y_i \in \mathbf{L}_x|\mathbf{x}) - \phi_g(y_j \in \mathbf{L}_x|\mathbf{x}). \tag{7}$$

Experiments validate this approximation and it works well on large scale training data.

3.3. Multiple reinitialization strategy

We use a multiple reinitialization strategy to calculate candidate predictions in different scale as well as their mirrors and combine them as final output. In the test phase, we augment the input image by resizing the image to 1.2 ×, 1.0 × and 0.8 × of original size, as well as its flipped version, all of which form the candidate set $\{s_i\}$. Each candidate image is processed by the trained model individually to get $|\{s_i\}| = 6$ corresponding unary maps and global scores.

We have tried average pooling and max pooling. Inspired by Lee et al. [27], we also try to combine them together,

$$\hat{s} = \frac{1}{2}(\text{avg}(\{s_i\}) + \max(\{s_i\})). \tag{8}$$

In practice, we find that learning to combine leads to a marginal improvement.

3.3.1. Learn to combine

We use DCNN to learn the weights to combine these score maps together. As the Caffe architecture is not efficient in processing different sizes of image at one iteration, we set an offline setup to train the weight parameters. All of these candidates s_i are computed beforehand by our FCN model. They are then added together by a corresponding weight λ_i

$$\hat{s} = \sum_i^N \lambda_i s_i, \tag{9}$$

where each λ_i is a vector in *#categories* dimensions (*e.g.*, *#categories* = 21 in Pascal VOC 2012 segmentation benchmark).

For back-propagating the gradient during training phase, we have

$$\frac{\partial L}{\partial \lambda_i} = \frac{\partial L}{\partial \hat{s}} * s_i. \tag{10}$$

Other parameters of the network are fixed in this training phase for computational efficiency.

3.4. Synchronized batch normalization

`BatchNorm` is an important component of the standard toolkit for training deep networks [17,28]. It is popular to train very deep and wide networks on distributed synchronized workers for large-scale image classification. `BatchNorm` is performed on the mini-batch of data on each GPU card independently, which will decrease the performance for small mini-batches. We implement the synchronized batch normalization. In synchronized batch normalization, each GPU card shares the same mean and variance in the forward passing. In the back propagation stage, the gradients with regard to the mean and variance of each batch are propagated, therefore synchronized `BatchNorm` on multi-GPU card yields to exactly the same results with one `BatchNorm`.

According to our experiments, both the *batch_size* of `BatchNorm` and the *batch_size* of SGD influence the final segmentation accuracy. With the synchronized batch norm we can freely control the *batch_size* of `BatchNorm` and SGD. We found large *batch_size* of `BatchNorm` cannot consistently improve the segmentation accuracy.

4. Experimental results

4.1. Implementation details

We use the public Caffe [29] framework and fine-tune the segmentation models on two back-bones including VGG-16 [16] and ResNet-101 [17].

Firstly on VGG-16, weight decay parameter is set to 0.0005. The momentum parameter is set to 0.99 and the initial learning rate is set at 1.0×10^{-5} as we process only one image at each iteration, *i.e.*, mini-batch size is set to 1 and crop size is 512. We apply "ploy" learning rate policy *base_lr* × $(1-\frac{iter}{max\_iter})^{0.9}$. We found this setting results in both increased accuracy and faster convergence by a little margin than 20 ∼ 30 images per min-batch [19]. We fine-tune the parameters of the classification network component from Fast R-CNN [30]. We get the FC6 and FC7 classifer layers from this detection net. This model is trained on the main part of Pascal VOC 2007 *trainvaltest* set with the main part of Pascal VOC 2012 *trainval* set with bounding box annotation on each image. None of the images in the *val* set and *test* set of Pascal VOC 2012 images appeared in the above image sets. The detection model is public available.[1] We found fine-tuned performance with this model is better than original VGG-16.

Secondly on ResNet-101, weight decay and momentum are set exactly to what they are in the pre-training stage on ImageNet-1k, which is 0.0001 and 0.9, respectively. The initial learning rate is 0.01 with the same aforementioned "ploy" learning rate. The crop size is 512 × 512 and the min-batch size is 16. We compare different batch size for BatchNorm and find $batch\_size = 8$ of BatchNorm leads to best performance. The difference of labeling rules between the original *train* set [31] and *aug* set [32] cannot be ignored in order to reach a high performance on this dataset. Therefore, it is necessary to further fine-tune the models on the original *train* set.

Algorithm 1 Synchronized Batch Normalization

```
Forward
input: $M$ GPU batch of data on M GPU card: $\mathcal{B}_i = \{x_{1...N}\}$,
       $i \in \{1...M\}$, each batch has $N$ samples;
       Layer Parameter: $\gamma$, $\beta$.

  1. For each GPU batch $\mathcal{B}_i$, $\mu_i = \frac{1}{N}\sum_{j=1}^{N} x_j$
  2. Synchronize batch mean, $\mu_{\mathcal{B}} = \frac{1}{M}\sum_{i=1}^{M} \mu_i$
  3. For each GPU batch $\mathcal{B}_i$, $\sigma_i^2 = \frac{1}{N}\sum_{j=1}^{N} (x_j - \mu_{\mathcal{B}})^2$
  4. Synchronize batch variance, $\sigma_{\mathcal{B}}^2 = \frac{1}{M}\sum_{i=1}^{M} \sigma_i^2$
  5. Normalize and reconstruct $y_i = \gamma\frac{x_i - \mu_{\mathcal{B}}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} + \beta$

output: $y_i$

Backward
input: Top gradient $\frac{\partial \ell}{\partial y_i}$; $M$ batch of data: $\mathcal{B}_i = \{x_{1...N}\}$;
       Layer Parameter: $\gamma$, $\beta$.

  1. For each GPU batch $\mathcal{B}_i$, $x_j \in \mathcal{B}_i$,
     $[\frac{\partial \ell}{\partial \gamma}]_i = \sum_{j=1}^{N} \frac{\partial \ell}{\partial y_j}$, $[\frac{\partial \ell}{\partial \beta}]_i = \sum_{j=1}^{N} \frac{\partial \ell}{\partial y_j}\hat{x}_j$
  2. Synchronize parameter gradients,
     $\frac{\partial \ell}{\partial \gamma} = \sum_{i=1}^{M} [\frac{\partial \ell}{\partial \gamma}]_i, \frac{\partial \ell}{\partial \beta} = \sum_{i=1}^{M} [\frac{\partial \ell}{\partial \beta}]_i$
  3. For each GPU batch $\mathcal{B}_i$, $x_j \in \mathcal{B}_i$,
     $\frac{\partial \ell}{\partial x_j} = \frac{\gamma}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}}\left(\frac{\partial \ell}{\partial y_j} - \frac{1}{MN}\left(\frac{\partial \ell}{\partial \gamma}\frac{(x_j - \mu_{\mathcal{B}}) - \frac{\mu_i - \mu_{\mathcal{B}}}{M}}{\sqrt{\sigma_{\mathcal{B}}^2 + \epsilon}} + \frac{\partial \ell}{\partial \beta}\right)\right)$

output: $\frac{\partial \ell}{\partial \gamma}$, $\frac{\partial \ell}{\partial \beta}$, $\frac{\partial \ell}{\partial x_j}$
```

Table 2
Comparison of different pooling strategies on the Pascal VOC 2012 *val set*. We learn to combine the estimation of average pooling and max pooling, which are two often used pooling methods. We found learning to combine them together is slightly better.

Strategy	Mean IoU(%)	Per-pixel acc
avg	74.09	0.9382
max	73.89	0.9376
(avg+max)/2	74.13	0.9384
Learn to combine	74.15	0.9385

4.2. Dataset

We have conducted experiments on three segmentation benchmark datasets.

Pascal VOC 2012 datasets [31] includes 20 categories plus background. The original *train* set has 1464 images with pixel-wise labels. We also use the annotation from [32], resulting in 10,582 (*trainaug* set), 1449 (*val* set) and 1456 (*test* set) images. Besides, we exploit the large scale segmentation set MS-COCO [33], which contains 123,287 images in its *trainval* set with 80 categories plus background. The accuracy is evaluated by mean IoU scores. Since the *test* set annotations of Pascal VOC 2012 is not released, the result on *test* set is reported by the server.[2]

Cityscapes dataset [34] consists of 2975 (*train*), 500 (*val*) and 1525 (*test*) pixel-wise labeled images and 19,998 *coarse* labeled images. There are 19 categories (object + stuff) in this datasets. Every image comes with the same size of 1024 × 2048 and all are about street scene in some European cities.

ADE20k dataset [35] is divided into 20,000 (*train*), 2000 (*val*) and 3000 (*test*) for training, validation and testing, respectively. There are 150 categories (object + stuff). ADE20k served as a competition dataset for scene parsing track in ILSVRC 2016.

4.3. Evaluation on different settings

We first evaluate different parameter settings in our model. We carry out experiments on Pascal VOC 2012 *val* set and the result is shown in Table 3. The first row is the mean IoU score of the Deeplab baseline, added by pool6 layers [10]. We pretrain the model on Microsoft COCO. We run 6 candidates and combine the segmentation score maps and the global scores by averaging.

The second row, "+clsNet", stands for the enhancement of the baseline net with a classification net. The performance is improved by about 1.5%, which verifies the effectiveness of global understanding provided by the classification net. The recognition accuracy of the categories, bottle, car, cow, plant and tv is increased more than 3%, while the categories of areo, bike, boat, mbike, person, sofa have less than 1% improvement or slight performance drop.

From the experiments, we also notice that the recognition ability of the classification network is the base for performance gains. In some rare cases when the global understanding gives wrong, the prediction of final segmentation map is even worse than the baseline.

The third row, "+compatibility", stands for the usage of global understanding for relation modeling. The fourth row, "+learn to combine", stands for the addition of different pooling strategies, with the comparison results shown in Table 2. Average pooling performs better than max pooling. By averaging the result maps of average pooling and max pooling, we get a slight improvement.

[1] https://github.com/rbgirshick/fast-rcnn.

[2] http://host.robots.ox.ac.uk:8080.

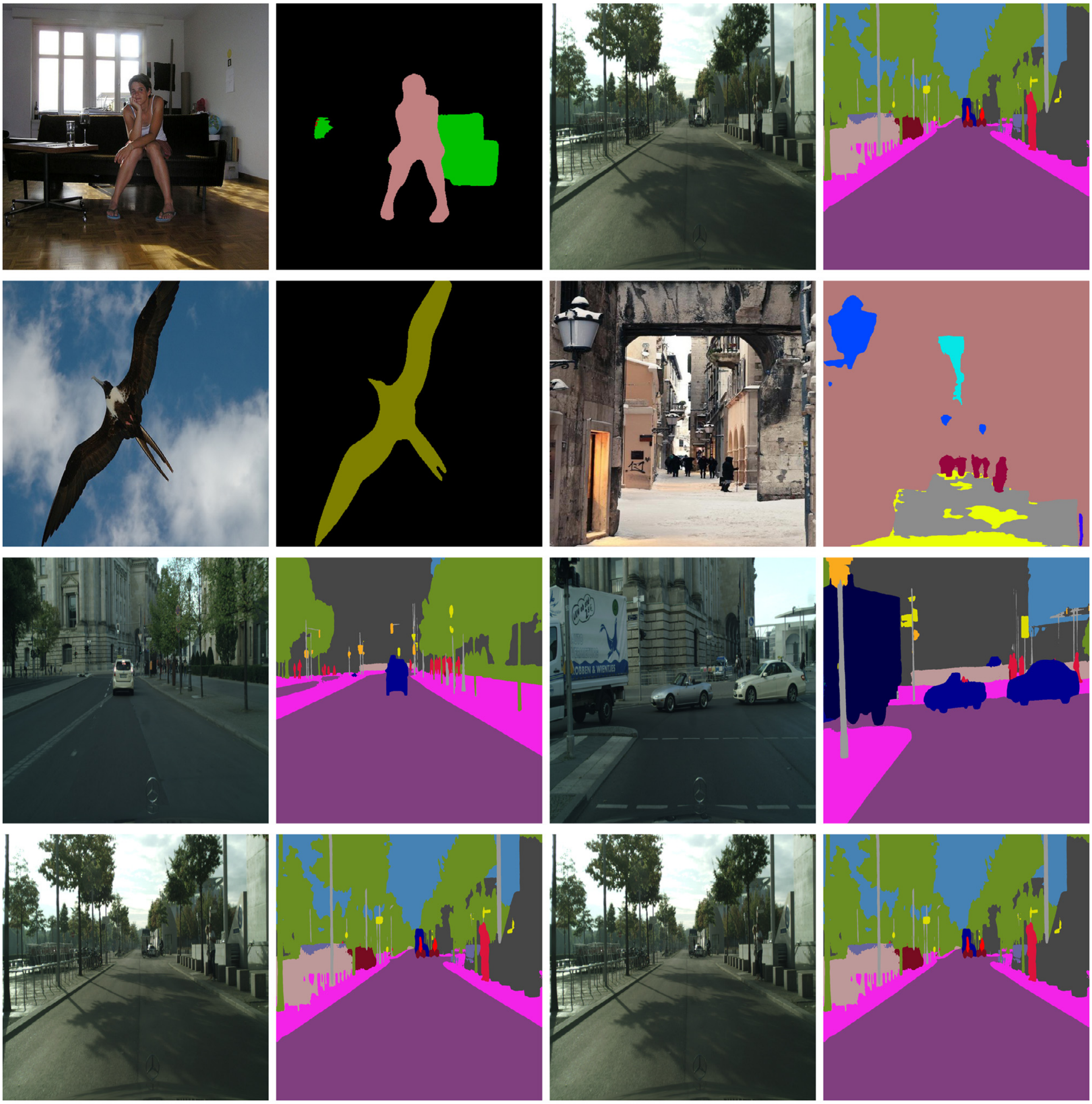


Fig. 3. Visual results on Pascal VOC 2012, Cityscapes and ADE20k.

Table 3
Labeling IoU (%) per category of our method on Pascal VOC 2012 *val* set (1449 images) under different setting. We use two back-bone models including VGG-16 and ResNet-101. The condition +label unary means that we use the multi-class labels for images from the validation set and perform prediction within the given categories. +label compatibility means we also use category co-occurrence in the label compatibilities matrix. Besides, if the ground truth labels are provided, it can reach a very high performance.

Method	VGG-16	ResNet-101
DeepLab baseline	72.54	76.35
+clsNet	74.13	77.71
+compatability	75.89	78.12
+learn to combine	75.99	81.32
+ground truth label	**79.58**	**83.21**

With learn-to-combine we further enhance the performance with another marginal improvement.

4.4. Comparisons with the state-of-the-art

ResNet-101 are widely used in semantic segmentation and object detection. We use ResNet-101 as backbone to compare with other methods in Table 4. Our method are competitive on Cityscapes and ADE-20k. It should be noted that DeepLabv3+_JFT [36] reaches a very high performance with the aid of inception back-bone model and very large extra dataset of JFT. Fig. 3 display some segmentation results from Pascal VOC 2012, Cityscapes and ADE-20k. Some failure cases are shown in Fig. 4. The segmentation models fail when the classification model misclassified the object categories.

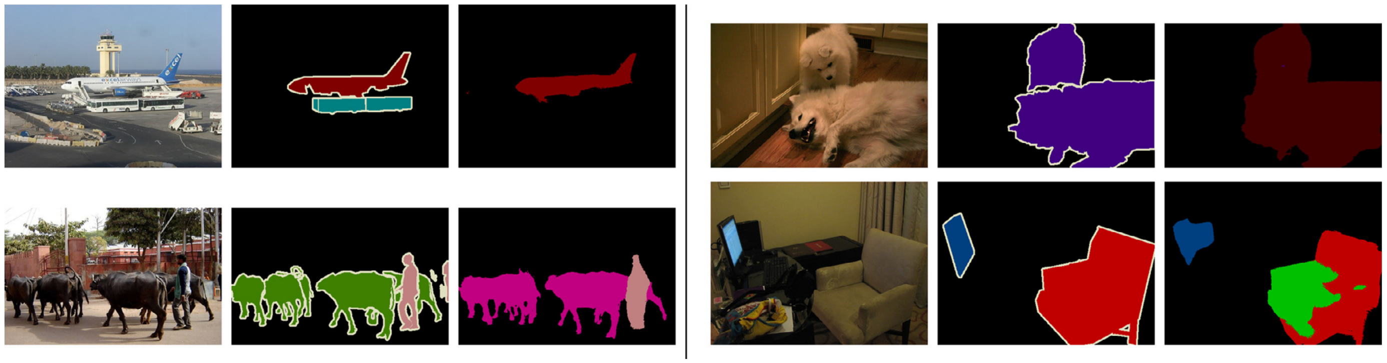


Fig. 4. Failure cases from the *val* set of the Pascal VOC 2012 dataset.

Table 4
Results of *single model* and *single scale* on the *val* sets on Pascal VOC 2012, Cityscapes and ADE20k. We have submitted *single model* and *multi scale* predictions to the corresponding servers to measure the performance on the *test* sets. The back-bone classification models for all these competing methods are ResNet-101 except for DeepLabv3+ [36]. Our models set $output\_stride = 16$ while other competing methods set $output\_stride = 8$, which means our methods costs about one-third computation of other methods.

Method	VOC 2012	CityScapes	ADE 20k
Single model and *multi scale* on *test* sets			
Adelaide [37]	79.1	–	–
deeplab-v2 [38]	79.7	–	–
HikSeg_COCO [39]	81.4	–	–
Large_kernel [40]	83.6	–	–
DUC-HDC [39]	–	77.6	–
VeryDeep [41]	79.1	74.6	–
DeepLabv3+_JFT [36]	**89.0**	**82.1**	–
Ours (ResNet-101, 16 ×)	81.4	81.4	**35.5**

5. Conclusion

We have proposed a joint segmentation and classification approach that exploits global image features for pixel-wise semantic segmentation. Although FCN has a relatively large receptive field, a global understanding to image content ensures more reasonable and reliable prediction. The unified classification and segmentation network have shown significant improvement in performance with slightly increased training time and computing resource. Besides, the proposed image-dependent label compatibility matrix models the co-occurrence among different labels, and it provides additional constraints for guidance CRF to obtain more accurate object boundaries.

References

[1] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, in: Proceedings of the 2012 Neural Information Processing Systems, 2012, pp. 1097–1105.

[2] H. Zhang, Y. Ji, W. Huang, L. Liu, Sitcom-star-based clothing retrieval for video advertising: a deep learning framework, Neural Comput. Appl. (2018) 1–20, doi:10.1007/s00521-018-3579-x.

[3] H. Zhang, X. Cao, J.K.L. Ho, T.W.S. Chow, Object-level video advertising: an optimization framework, IEEE Trans. Ind. Inf. 13 (2) (2017) 520–531.

[4] H. Zhang, S. Wang, X. Xu, T.W.S. Chow, Q.M.J. Wu, Tree2vector: learning a vectorial representation for tree-structured data, IEEE Trans. Neural Netw. Learn. Syst. PP (99) (2018) 1–15.

[5] Y. LeCun, Y. Bengio, Convolutional networks for images, speech, and time series, in: The Handbook of Brain Theory and Neural Networks, 3361, MIT Press, 1995.

[6] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al., Imagenet large scale visual recognition challenge, Int. J. Comput. Vis. 115 (3) (2015) 211–252.

[7] F. Shen, R. Gan, S. Yan, G. Zeng, Semantic segmentation via structured patch prediction, context CRF and guidance CRF, in: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1953–1961.

[8] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, P. Torr, Conditional random fields as recurrent neural networks, in: Proceedings of the 2015 International Conference on Computer Vision (ICCV), 2015.

[9] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang, P. Torr, Conditional random fields as recurrent neural networks, in: Proceedings of the 2015 International Conference on Computer Vision (ICCV), 2015.

[10] W. Liu, A. Rabinovich, A.C. Berg, Parsenet: Looking wider to see better. arXiv preprint arXiv:1506.04579 (2015).

[11] G. Papandreou, L.-C. Chen, K. Murphy, A.L. Yuille, Weakly- and Semi-Supervised Learning of a DCNN for Semantic Image Segmentation. arXiv preprint arXiv:1502.02734 (2015).

[12] Z. Liu, X. Li, P. Luo, C.C. Loy, X. Tang, Semantic image segmentation via deep parsing network, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1377–1385.

[13] H. Noh, S. Hong, B. Han, Learning deconvolution network for semantic segmentation, in: Proceedings of the IEEE International Conference on Computer Vision, 2015, pp. 1520–1528.

[14] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, Y. LeCun, Overfeat: Integrated recognition, localization and detection using convolutional networks. arXiv preprint arXiv:1312.6229 (2013).

[15] J. Long, E. Shelhamer, T. Darrell, Fully convolutional networks for semantic segmentation, in: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, 2015 arXiv:1411.4038.

[16] K. Simonyan, A. Zisserman, Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014).

[17] K. He, X. Zhang, S. Ren, J. Sun, Deep residual learning for image recognition, in: Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 770–778.

[18] P. Krähenbühl, V. Koltun, Efficient inference in fully connected CRFs with Gaussian edge potentials, in: Proceedings of the 2012 European Conference on Computer Vision, ECCV, 2012.

[19] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, Semantic image segmentation with deep convolutional nets and fully connected CRFs, in: Proceedings of the 2015 International Conference on Learning Representations, ICLR, 2015 arXiv:1412.7062.

[20] G. Lin, C. Shen, A. Van Den Hengel, I. Reid, Efficient piecewise training of deep structured models for semantic segmentation, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3194–3203.

[21] P. Krähenbühl, V. Koltun, Parameter learning and convergent inference for dense random fields, in: Proceedings of the Thirtieth International Conference on Machine Learning (ICML-13), 2013, pp. 513–521.

[22] A.G. Schwing, R. Urtasun, Fully connected deep structured networks. arXiv preprint arXiv:1503.02351 (2015).

[23] L.-C. Chen, A.G. Schwing, A.L. Yuille, R. Urtasun, Learning deep structured models., in: Proceedings of the 2015 International Conference on Machine Learning, ICML, in: JMLR Proceedings, 37, 2015, pp. 1785–1794. http://dblp.uni-trier.de/db/conf/icml/icml2015.html#ChenSYU15. JMLR.org.

[24] G. Lin, C. Shen, I.D. Reid, A. van den Hengel, Deeply learning the messages in message passing inference., in: Proceedings of the 2015 Neural Information Processing Systems (NIPS), 2015.

[25] Y. Wei, W. Xia, J. Huang, B. Ni, J. Dong, Y. Zhao, S. Yan, CNN: single-label to multi-label (2014). arXiv preprint arXiv:1406.5726.

[26] Y. Gong, Y. Jia, T. Leung, A. Toshev, S. Ioffe, Deep convolutional ranking for multilabel image annotation. arXiv preprint arXiv:1312.4894 (2013).

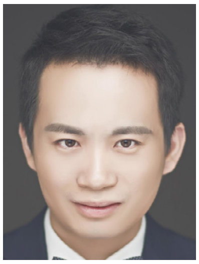
[27] C.-Y. Lee, P.W. Gallagher, Z. Tu, Generalizing pooling functions in convolutional neural networks: Mixed, gated, and tree, in: Artificial Intelligence and Statistics, 2016, pp. 464–472.

[28] S. Ioffe, C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint 1502.03167 (2015).

[29] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, T. Darrell, Caffe: convolutional architecture for fast feature embedding, in: Proceedings of the 2014 ACM International Conference on Multimedia, ACM, 2014, pp. 675–678.

[30] R. Girshick, Fast R-CNN, in: Proceedings of the 2015 International Conference on Computer Vision (ICCV), 2015.

[31] M. Everingham, L. Van Gool, C.K. Williams, J. Winn, A. Zisserman, The pascal visual object classes (VOC) challenge, Int. J. Comput. Vis. 88 (2) (2010) 303–338.
[32] B. Hariharan, P. Arbeláez, L. Bourdev, S. Maji, J. Malik, Semantic contours from inverse detectors, in: Proceedings of the 2011 IEEE International Conference on Computer Vision (ICCV), IEEE, 2011, pp. 991–998.
[33] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C.L. Zitnick, Microsoft COCO: common objects in context, in: Proceedings of the European Conference on Computer Vision, ECCV 2014, Springer, 2014, pp. 740–755.
[34] M. Cordts, M. Omran, S. Ramos, T. Rehfeld, M. Enzweiler, R. Benenson, U. Franke, S. Roth, B. Schiele, The cityscapes dataset for semantic urban scene understanding, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2016, pp. 3213–3223.
[35] B. Zhou, H. Zhao, X. Puig, S. Fidler, A. Barriuso, A. Torralba, Scene parsing through ADE20K dataset, in: Proceedings of the 2017 Conference on Computer Vision and Pattern Recognition, CVPR, 2017.
[36] L.-C. Chen, Y. Zhu, G. Papandreou, F. Schroff, H. Adam, Encoder–Decoder with Atrous Separable Convolution for Semantic image Segmentation, arXiv preprint:1802.02611(2018).
[37] Z. Wu, C. Shen, A.v. d. Hengel, High-Performance Semantic Segmentation Using Very Deep Fully Convolutional Networks, arXiv preprint:1604.04339 (2016).
[38] L.-C. Chen, G. Papandreou, I. Kokkinos, K. Murphy, A.L. Yuille, DeepLab: semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs, IEEE Trans. Pattern Anal. Mach. Intell. 40 (4) (2018) 834–848.
[39] P. Wang, P. Chen, Y. Yuan, D. Liu, Z. Huang, X. Hou, G. Cottrell, Understanding Convolution for Semantic Segmentation, arXiv preprint:1702.08502 (2017).
[40] C. Peng, X. Zhang, G. Yu, G. Luo, J. Sun, Large Kernel Matters – Improve Semantic Segmentation by Global Convolutional Network, arXiv preprint:1703.02719 (2017).
[41] Z. Wu, C. Shen, A.v. d. Hengel, Bridging Category-Level and Instance-Level Semantic Image Segmentation, arXiv preprint:1605.06885 (2016).



Falong Shen obtain his B.S. degree from School of Earth and Space Science at Peking University in 2013, and currently is a Ph.D. candidate in Department of Electronic Engineering and Computer Sciences at Peking University. He has published two first-author papers on CVPR. He joined the scene parsing track of ILSVRC competitions and was invited to give a poster at ECCV 2016. His research interests include semantic image segmentation, neural style transfer, generative adversarial networks and high-performance programming.

Gang Zeng obtained his B.S. degree from School of Mathematical Sciences at Peking University in 2001, and his Ph.D. degree under the supervision of Prof. Long Quan at Computer Vision and Graphics Group in Department of Computer Science and Engineering at Hong Kong University of Science and Technology in 2006. By the end of 2017, he has published 40 peer-reviewed papers, including 19 papers in top two computer vision conferences (*i.e.*, CVPR, ICCV) and top two computer vision journals (*i.e.*, PAMI, IJCV), and 3 papers in SIGGRAPH. The h-index is 16 and the citations reaches 1210 according to Google Scholar.